US006178507B1

(12) United States Patent
Vanstone

(10) Patent No.: US 6,178,507 B1
(45) Date of Patent: Jan. 23, 2001

(54) **DATA CARD VERIFICATION SYSTEM**

(75) Inventor: **Scott A Vanstone**, Waterloo (CA)

(73) Assignee: **Certicom Corp.**, Ontario (CA)

(*) Notice: Under 35 U.S.C. 154(b), the term of this patent shall be extended for 0 days.

(21) Appl. No.: **09/016,926**

(22) Filed: **Feb. 2, 1998**

(30) **Foreign Application Priority Data**

Feb. 3, 1997 (GB) 9702152

(51) **Int. Cl.**^{7} **H04L 9/00**

(52) **U.S. Cl.** **713/169**; 380/43; 380/259; 380/283; 380/285; 705/67; 713/168; 713/172; 713/176; 713/180

(58) **Field of Search** 380/21, 23, 24, 380/30, 44, 283, 285; 705/64, 65, 67, 71, 73; 713/168, 169, 172, 176, 180

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,748,668		5/1988	Shamir et al.	380/29
4,890,323		12/1989	Beker et al.	380/28
4,995,082	*	2/1991	Schnorr	380/23
5,218,637		6/1993	Angebaud et al.	380/25
5,299,263	*	3/1994	Beller et al.	380/30
5,400,403		3/1995	Fahn et al.	380/21
5,406,628	*	4/1995	Beller et al.	380/30
5,627,893		5/1997	Demytko	380/30
5,721,781	*	2/1998	Deo et al.	380/25
5,748,740	*	5/1998	Curry et al.	380/25
5,793,866		8/1998	Brown et al.	380/2
5,805,702	*	9/1998	Curry et al.	380/24
5,825,880		10/1998	Sudia et al.	380/21
5,870,470		2/1999	Johnson et al.	380/6
5,881,038		3/1999	Oshima et al.	369/59
5,907,618		5/1999	Gennaro et al.	380/21
5,917,913		6/1999	Wang	380/25
5,955,717		9/1999	Vanstone	235/380
5,960,084		9/1999	Angelo	380/25
6,038,549		3/2000	Davis et al.	705/35
6,041,314		3/2000	Davis	705/41

FOREIGN PATENT DOCUMENTS

0 588 339	3/1994	(EP) .
2 536 928	6/1984	(FR) .
WO 91/16691	10/1991	(WO) .

OTHER PUBLICATIONS

Schneier, Bruce, Applied Cryptography, 1996, pp.35–36.

Miyaji, A: "Elliptic Curves Suitable For Cryptosystem", IEICE Transactions On Fundamentals of Electronics, Communications and Computer Sciences, vol. E77–A, No.1, Jan. 1, 1994, pp. 98–104, XP000439669.

Schnorr, C P: "Efficcient Signature Generation By Smart Cards", Journal of Cryptology, vol. 4, No. 3, Jan. 1, 1991, pp. 161–174, XP000574352.

Kenji, Koyoma et al: "Elliptic Curve Cryptosystems And Their Applications", IEICE Transactions On Information And Systems, vol. E75–D, No. 1, Jan. 1, 1992, pp. 50–57, XP000301174.

Waleffe, D De et al: "Corsair: A Smart Card For Public Key Cryptosystems", Advances In Cryptology—Proceedings of Crypto, Santa Barbara, Aug. 11–15, 1990, No. CONF. 10, Jan. 1, 1990, pp. 502–513, XP000260013, Menezes, A J; Vanstone, S A.

Koblitz, N: "Elliptic Curve Cryptosystems", Mathematics Of Computation, vol. 48, No. 177, Jan. 1987, pp. 203–209, XP000671098.

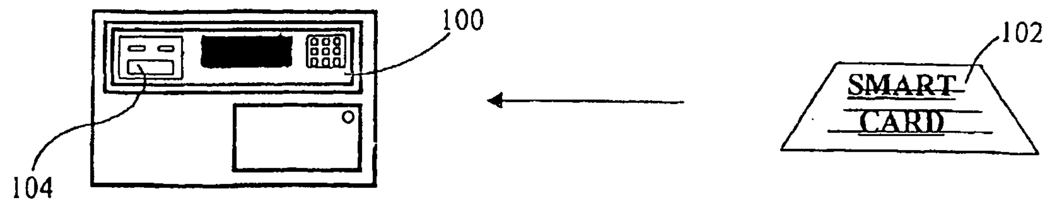
* cited by examiner

Primary Examiner—Tod R. Swann
Assistant Examiner—Paul E. Callahan
(74) *Attorney, Agent, or Firm*—Finnegan, Henderson, Farabow, Garrett, & Dunner, L.L.P.

(57) **ABSTRACT**

A method for verifying the authenticity of messages exchanged between a pair of correspondents in an electronic conducted over a data transmission system where the correspondents each include respective signing and verifying portions of a first signature scheme and a second signature scheme different from the first and utilizing an elliptic curve cryptosystem.

8 Claims, 2 Drawing Sheets



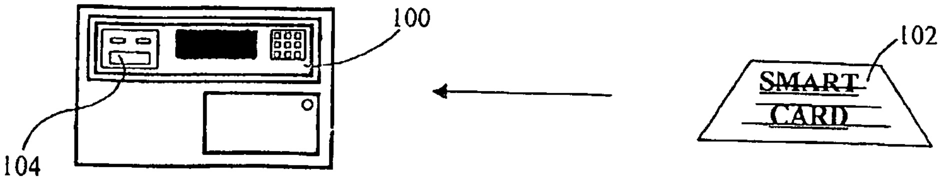


Figure 1a

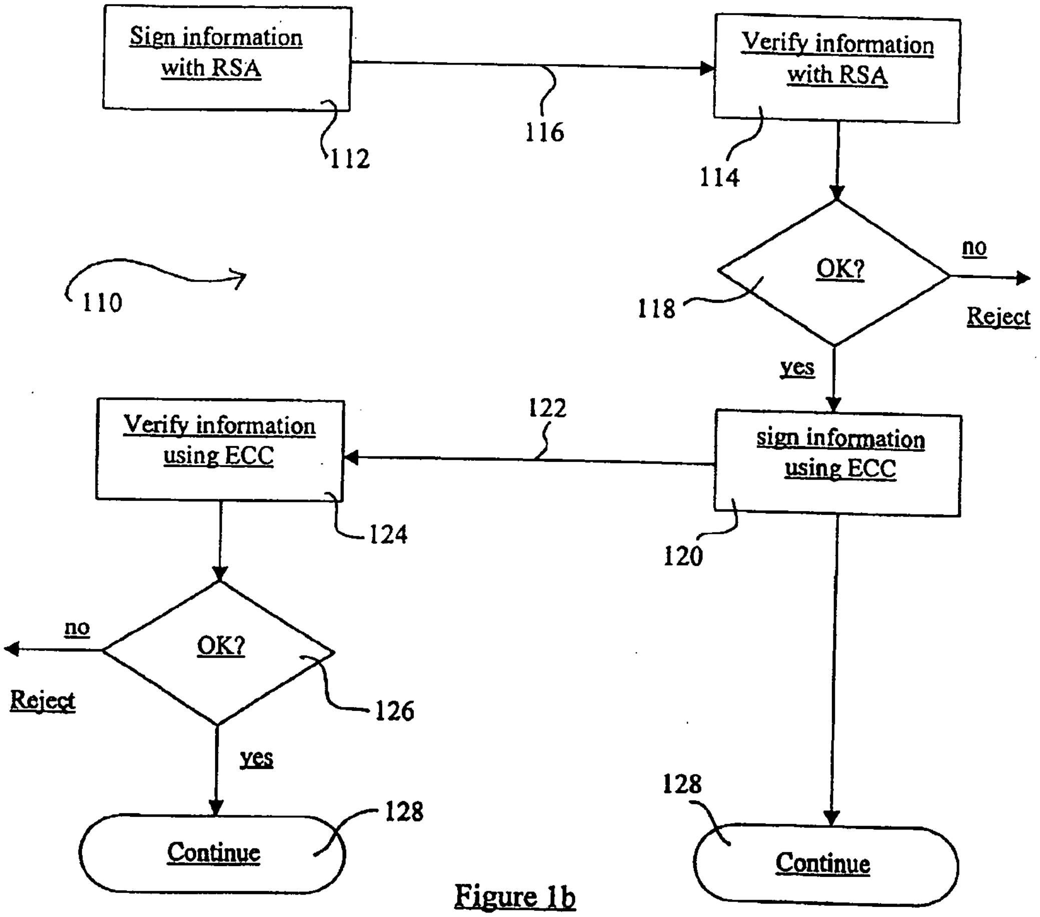

Sign information with RSA
112
116
Verify information with RSA
114
110
OK?
118
no
Reject
yes
122
Verify information using ECC
124
sign information using ECC
120
no
OK?
Reject
126
yes
128
Continue
128
Continue

Figure 1b

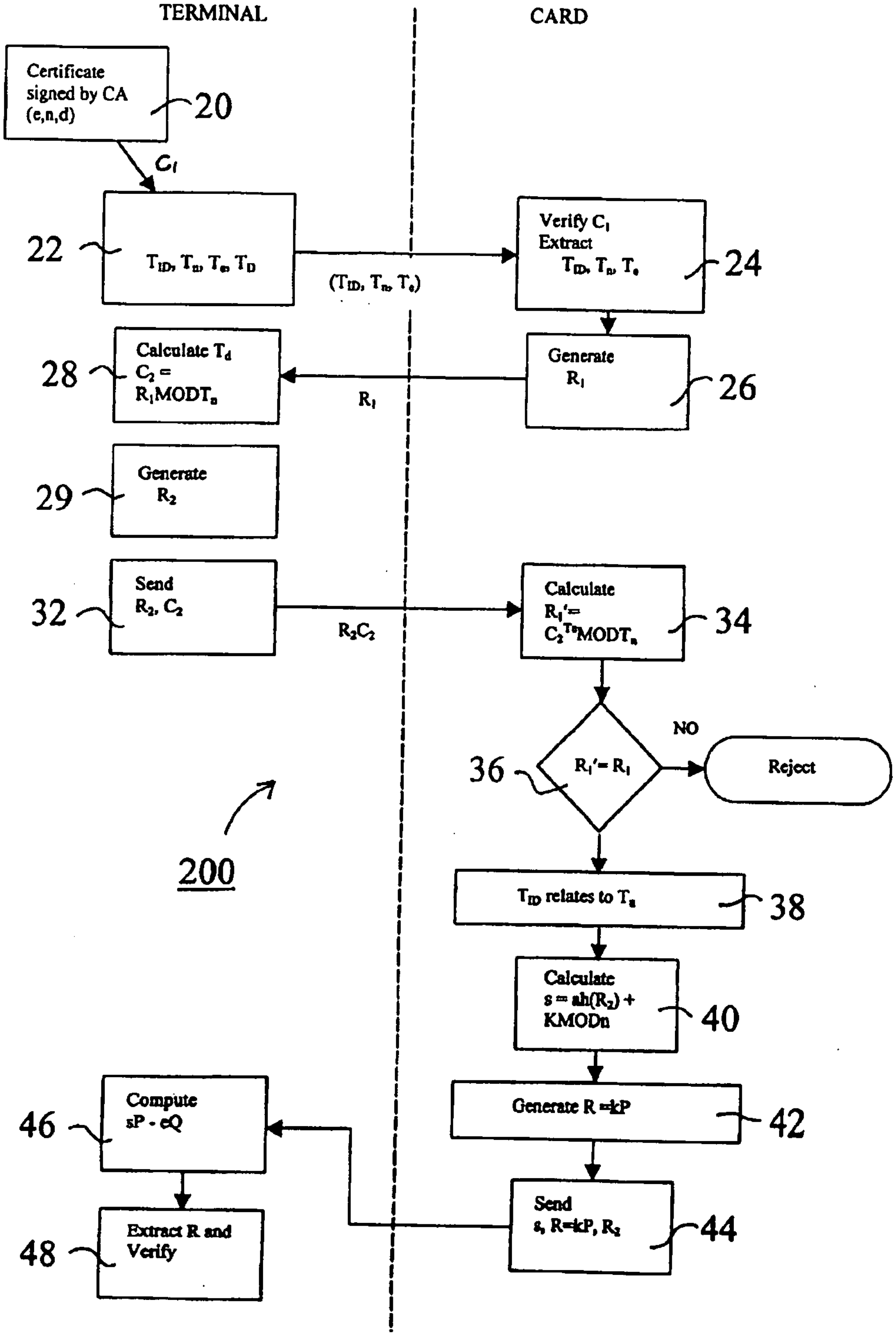

TERMINAL
CARD
Certificate signed by CA (e,n,d)
20
C_1
22
T_{ID}, T_n, T_e, T_D
(T_{ID}, T_n, T_e)
Verify C_1 Extract T_{ID}, T_n, T_e
24
28
Calculate T_d C_2 = R_1MODT_n
R_1
Generate R_1
26
29
Generate R_2
32
Send R_2, C_2
R_2C_2
Calculate R_1'= C_2^{Te}MODT_n
34
36
R_1'= R_1
NO
Reject
T_{ID} relates to T_e
38
Calculate s = ah(R_2) + KMODn
40
Generate R =kP
42
46
Compute sP - eQ
Send s, R=kP, R_2
44
48
Extract R and Verify
200

FIGURE 2

DATA CARD VERIFICATION SYSTEM

This invention relates to methods and apparatus for data transfer and authentication in an electronic transaction system, and more particularly to electronic transaction systems utilizing smart cards.

BACKGROUND OF THE INVENTION

It has become widely accepted to conduct transactions such as financial transactions or exchange of documents electronically. Automated teller machines (ATMs) and credit cards are widely used for personal transaction and as their use expands so too does the need to verify such transactions increase. A smart card is somewhat like a credit card and includes some processing and storage capability. Smart cards are prone to fraudulent misuse, for example by a dummy terminal which is used to glean information from an unsuspecting user. Thus, before any exchange of critical information takes place between either a terminal and a smart card or vice versa it is necessary to verify the authenticity of the terminal as well as the card. One of these verifications may take the form of "signing" an initial transaction digitally so that the authenticity of the transaction can be verified by both parties involved in the subsequent session. The signature is performed according to a protocol that utilizes a random message, i.e. the transaction and a secret key associated with the party.

The signature must be performed such that the party's secret key cannot be determined. To avoid the complexity of distributing secret keys, it is convenient to utilize a public key encryption scheme in the generation of the signature. Such capabilities are available where the transaction is conducted between parties having access to relatively large computing resources, but it is equally important to facilitate such transactions at an individual level where more limited computing resources available, as in the smart card.

Transaction cards or smart cards are now available with limited computing capacity, but these are not sufficient to implement existing digital signature protocols in a commercially viable manner. As noted above, in order to generate a verification signature it is necessary to utilize a public key inscription scheme. Currently, most public key schemes are based on RSA, but the DSS and the demand for a more compact system are rapidly changing this. The DSS scheme, which is an implementation of a Diffie-Hellman public key protocol, utilizes the set of integers Z_p where p is a large prime. For adequate security, p must be in the order of 512 bits, although the resultant signature may be reduced mod q, where q divides p−1, and may be in the order of 160 bits.

An alternative encryption scheme which was one of the first fully fledged public key algorithms and which works for encryption as well as for digital signatures is known as the RSA algorithm. RSA gets it security from the difficulty of factoring large numbers. The public and private keys are functions of a pair of large (100 to 200 digits or even larger) of prime numbers. The public key for RSA encryption is n, the product of the two primes p and q where p and q must remain secret and e which is relatively prime to $(p-1)\times(q-1)$. the encryption key d is equal to e^{-1} $(\mathrm{mod}(p-1)\times(q-1))$. Note that d and n are relatively prime.

To encrypt a message m, first divide into a number of numerical blocks such that each block is a unique representation modulo n, then the encrypted message block c_i is simply m_i^c (mod n). To decrypt a message take each encrypted block c_i and compute $m_i=c_i^d$ (mod n).

Another encryption scheme that provides enhanced security at relatively small modulus is that utilizing elliptic curves in the finite field 2^m. A value of m in the order of 155 provides security comparable to a 512 bit modulus DSS and therefore offers significant benefits in implementation.

Diffie-Hellman public key encryption utilizes the properties of discrete logs so that even if a generator β and the exponentiation β^k is known, the value of k cannot be determined. A similar property exist with elliptic curves where the addition of two points on any curve produces a third point on the curve. Similarly, multiplying a point P on the curve by an integer k produces a further point on the curve. For an elliptic curve, the point kP is simply obtained by adding k copies of the point P together.

However, knowing the starting point and the end point does not reveal the value of the integer k which may then be used as a session key for encryption. The value kP, where P is an initial known point is therefore equivalent to the exponentiation β^k. Furthermore, elliptic curve crypto-systems offer advantages over other key crypto-systems when bandwidth efficiency, reduced computation and minimized code space are application goals.

Furthermore, in the context of a smart card and an automated teller machine transaction, there are two major steps involved in the authentication of both parties. The first is the authentication of the terminal by the smart card and the second is the authentication of the smart card by the terminal. Generally, this authentication involves the verification of a certificate generated by the terminal and received by the smart card and the verification of a certificate signed by the smart card and verified by the terminal. Once the certificates have been positively verified the transaction between the smart card and the terminal may continue.

Given the limited processing capability of the smart card, verifications and signature processing performed on the smart card are generally limited to simple encryption algorithms. A more sophisticated encryption algorithm is generally beyond the scope of the processing capabilities contained within the smart card. Thus, there exist a need for a signature verification and generation method which may be implemented on a smart card and which is relatively secure.

SUMMARY OF THE INVENTION

This invention seeks in one aspect to provide a method of data verification between a smart card and a terminal.

In accordance with this aspect there is provided a method for verifying a pair of participants in an electronic transaction, comprising the steps of verifying information received by the second participant from the first participant, wherein the verification is performed according to a first signature algorithm; verifying information received by the first participant from the second participant, wherein the verification is performed according to a second signature algorithm; and whereby the transaction is rejected if either verification fails.

The first signature algorithm may be one which is computationally more difficult in signing than verifying, while the second signature algorithm is more difficult in verifying than signing. In such an embodiment the second participant may participate with relatively little computing power, while security is maintained at a high level.

In a further embodiment, the first signature algorithm is based on an RSA, or DDS type algorithm, and the second signature algorithm is based on an elliptic curve algorithm.

BRIEF DESCRIPTION OF THE DRAWINGS

An embodiment of the invention will now be described by way of example on the reference to the accompanying drawings, in which, FIG. 1*a* is a schematic representations showing a smart card and terminal;

FIG. 1*b* is a schematic representations showing the sequence of events performed during the verification process in a smart card transaction system; and

FIG. 2 is a detailed schematic representation showing a specific protocol.

DETAILED DESCRIPTION OF A PREFERRED EMBODIMENT

Referring to FIG. 1(*a*), a terminal **100** is adapted to receive a smart card **102**. Typically, insertion of the card **102** into the terminal initiates a transaction. Mutual authentication between the terminal and the card is then performed as shown in FIG. 1*b*. In very general terms, this mutual authentication is performed according to a "challenge-response" protocol. Generally, card transmits information to the terminal, the terminal **100** signs information with an RSA based algorithm **112** and is then sent to the card **102**, which verifies the information with an RSA based algorithm **114**. The information exchange **116** between the card and the terminal also includes information generated by the card which is sent to the terminal to be signed by the terminal with an RSA algorithm and returned to the card to be verified utilizing a RSA algorithm. Once the relevant verification has been performed **118**, a further step is performed where information is signed by the card using an elliptic curve encryption protocol **120** and submitted to the terminal to be verified **124** by the terminal utilizing an elliptic curve based protocol. Similarly, the information exchange **122** between the card and the terminal may include information generated by the terminal which is sent to the card to be signed by the card and returned to the terminal for verification. Once the appropriate information has been verified **126** the further transactions between the terminal and card may proceed **128**.

Referring now to FIG. **2**, a detailed implementation of the mutual authentication of the terminal and the card, according to the "challenged-response" protocol is shown generally by numeral **200**. The terminal **100** is first verified by the card **102** and the card is then verified by the terminal. The terminal first sends to the card a certificate C_1, **20** containing its ID, T_{ID}, and public information including the public key. The certificate **20** may be also signed by a certifying authority (CA) so that the card may verify the association of the terminal ID T_{ID} with the public key received from the terminal. The keys used by the terminal and the CA in this embodiment may both be based on the RSA algorithm.

With the RSA algorithm each member or party has a public and a private key, and each key has two parts. The signature has the form:

$$S=m^d (mod\ n)$$

where:

m is the message to be signed;

n a public key is the modulus and is the product of two primes p and q;

e the encryption key chosen at random and which is also public is a number chosen to be relatively prime to (p−1)×(q−1); and

d the private key which is congruent to e^{-1} (mod(p−1)×(q−1)).

For the RSA algorithm, the pair of integers (n,e) are the public key information that is used for signing. While, the pair of integers (d,n) may be used to decrypt a message which has been encrypted with the public key information (n,e).

Referring back to FIG. **2**, the numbers n and e are the public keys of the CA and may be set as system parameters. The public key e may be either stored in the smart card or in an alternate embodiment hardwired into an logic circuit in the card. Furthermore, by choosing e to be relatively small, ensures that the exponentiation may be carried out relatively quickly.

The certificate **20** C_1 is signed by the CA and has the parameters (n,e). The certificate contains the terminal ID T_{Id}: and the terminal public key information T_n and T_e which is based on the RSA algorithm. The certificate C_1 is verified **24** by the card extracting T_{ID}, T_n, T_e. This information is simply extracted by performing C_1^e mod n. The card then authenticates the terminal by generating a random number **R1**, **26**, which it transmits to the terminal. The terminal signs the message R1 using its secret key T_d by performing $R1^{T_e}$ $MODT_n$ to generate the value C_2, **28**. Once again the key used by the terminal is an RSA key which has been originally created in such a way that the public key T_e consist of a small possibly system wide parameter having a value 3, while the other part of the public key is the modulus T_n which would be associated with the terminal. The terminals private key T_d cannot be small if it corresponds to a small public key T_e. In the case of the terminal, it does not matter whether the private key T_d is chosen to be large as the terminal has the required computing power to perform the exponentiation relative quickly.

Once the terminal has calculated the value C_2, **28**, it generates a secret random number R2, **29** the terminal sends both R2 and C_2, **32** to the card. The card then performs a modular exponentiation **34** on the signed value C_2 with the small exponent T_e, using the terminal's modulus T_n. This is performed by calculating $R1'=C_2^{Te}$ mod T_n. If R1' is equal to R1, **36** then the card knows that it is dealing with the terminal whose ID T_{ID} is associated **38** with the modulus T_n. The card generally contains a modulo arithmetic processor (not shown) to perform the above operation.

The secret random number R2 is signed **40** by the card and returned to the terminal along with a certificate signed by the CA which relates the card ID to its public information. The signing by the card is performed according to an elliptic curve signature algorithm.

The verification of the card proceeds on a similar basis as the verification of the terminal, however, the signing by the card utilizes an elliptic curve encryption system.

Typically for an elliptic curve implementation a signature component s has the form:

$$s=ae+k(mod\ n)$$

where:

P is a point on the curve which is a predefined parameter of the system;

k is a random integer selected as a short term private or session key, and has a corresponding short term public key R=kP;

a is the long term private key of the sender (card) and has a corresponding public key aP=Q;

e is a secure hash, such as the SHA hash function, of a message m (R2 in this case) and short term public key R; and

n is the order of the curve.

For simplicity it will be assumed that the signature component s is of the form s=ae+k as discussed above although it will be understood that other signature protocols may be used.

To verify the signature sP−eQ must be computed and compared with R. The card generates R, using for example a field arithmetic processor (not shown). The card sends to the terminal a message including m, s, and R, indicated in block 44 of FIG. 2 and the signature is verified by the terminal by computing the value (sP–eQ) 46 which should correspond to kP. If the computed values correspond 48 then the signature is verified and hence the card is verified and the transaction may continue.

The terminal checks the certificate, then it checks the signature of the transaction data which contains R2, thus authenticating the card to the terminal. In the present embodiment the signature generated by the card is an elliptic curve signature, which is easier for the card to generate, but requires more computation by the terminal to verify.

As is seen from the above equation, the calculation of s is relatively straightforward and does not require significant computing power. However in order to perform the verification it is necessary to compute a number of point multiplications to obtain sP and eQ, each of which is computationally complex. Other protocols, such as the MQV protocols require similar computations when implemented over elliptic curves which may result in slow verification when the computing power is limited. However this is generally not the case for a terminal.

Although an embodiment of the invention has been described with reference to a specific protocol for the verification of the terminal and for the verification of the card, other protocols may also be used.

What is claimed is:

1. A method of verifying the authenticity of messages exchanged between a pair of correspondents in an electronic transaction conducted over a data transmission system, said correspondents each including respective signing and verifying portions of a first signature scheme and a second signature scheme different to said first scheme and utilizing an elliptic curve crypto system said method comprising the steps of:

one of said correspondents signing a message according to a signing portion of one of said schemes associated with said one correspondent to provide a first signed message and transmitting said first signed message to another of said correspondents; said other correspondent utilizing said verifying portion of said one signature scheme to verify said first signed message received from said one correspondent;

said other correspondent signing a message by utilizing said signing portion of the other of said signature schemes to provide a second signed message and transmitting a second signed message to said one correspondent;

said one correspondent verifying said second signed message received from said other correspondent by utilizing said verification portion of said other of said signature schemes, wherein one of said signature and one of said verifications is performed according to said second signature scheme utilizing an elliptic curve cryptosystem; and rejecting said transaction if either verification fails.

2. A method as defined in claim 1, said first signature scheme is computationally more difficult in signing than verifying, while said second signature scheme is computationally more difficult in verifying than signing, thereby allowing one of said correspondents to participate with relatively little computing power while maintaining security of said transaction.

3. A method as defined in claim 1, wherein said first digital signature scheme is an RSA type scheme.

4. A method as defined in claim 1, wherein said first digital signature scheme is a DSS type scheme.

5. A method of verifying the authenticity of messages exchanged between a pair of correspondents in electronic transaction conducted over a data transmission system, said correspondents each including respective signing and verifying portions of a first signature scheme and a second signature scheme, different from said first scheme and utilizing an elliptic curve crypto system said method comprising the steps of:

one of said correspondents transmitting to another of said correspondents, a first certificate including public key and identification information of said first correspondent;

said other correspondent verifying said certificate and extracting said public key said identification information therefrom;

said other correspondent generating a first challenge R_1 and transmitting said challenge to said one correspondent;

said one correspondent signing said received challenge R_1 in accordance with said signing portion of one of said signature schemes to provide a second certificate C2;

said one correspondent generating a second challenge and transmitting said second challenge along with said certificate C2 to said other correspondent;

said other correspondent verifying said certificate C2 in accordance with said verification portion of one of said signature schemes;

said other correspondent signing said second challenge R2 in accordance with said signing portion of the other of said signature schemes to provide a third certificate and transmitting said said third certificate to said one correspondent; and

said one correspondent verifying said third certificate in accordance with said verification portion of said other of said signature schemes, and rejecting said transaction if either said signature is not verified.

6. A smart card for use in an electronic transaction with a correspondent, said card comprising:

a memory including

a verification algorithm of a first signature scheme to implement a verification of a signature performed according to a first signature generation algorithm by said correspondent;

a signing algorithm of second signature scheme different to said first signature scheme and utilizing elliptic curve cryptography, said algorithm implementing a signature according to a second signature generation algorithm;

a program for invoking said algorithms; and

processor means for running said first verification algorithm for verifying a first message signed by sad correspondent and for running said second signature for signing a second message for transmission to said correspondent.

7. A card according to claim 6 wherein said verification algorithm verifies an RSA signature.

8. A card according to claim 6 wherein said verification algorithm verifies a DSS signature.

* * * * *